IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Mohammad R. Haghighat, et al. § Group Art Unit: 2168
§
Serial No.: 10/685,070 §
§ Examiner: Dangelino N. Gortayo
Filed: October 13, 2003 §
§
For: Concurrent Insertion Of Elements Into Data Structures § Atty. Dkt. No.: ITL.2266US (P17567)
§

Mail Stop **Appeal Brief-Patents**
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REPLY BRIEF

Sir:

Appellants respectfully file this Reply Brief in response to the Examiner's Answer mailed on March 22, 2010.

I. REPLY

The Examiner contends, apparently, that determining whether an existing data element is not a member of a data structure, part of the recited subject matter of claim 13, is met by column 19, lines 6-25 of Garthwaite. Examiner's Answer, page 9. It is respectfully submitted that this is incorrect. Garthwaite in this portion simply teaches that certain bits of a table identifier field are used as a tag to indicate whether a remembered set structure is in the process of being replaced by another structure. This has nothing to do with the recited subject matter, namely determining whether an existing data element is not a member of a data structure.

Instead, Garthwaite simply teaches that based on the given tag value, it can be determined whether no replacement operation is underway, a replacement is underway but an old remembered set location can be found, or a replacement is underway and a list of insertion

Date of Deposit: May 21, 2010
I hereby certify under 37 CFR § 1.8 this correspondence is being deposited **via EFS** on the date indicated above.

/Stephanie Petreas/
Stephanie Petreas

records that have not been inserted into a remembered set table can be identified. None of these instances, however, have anything to do with the recited subject matter of determining whether an existing data element is not a member of a data structure. Instead, all that Garthwaite teaches here is that based on a tag value, a status of the replacement of a remembered set structure can be determined.

Still as to claim 13, as contended support for the recited subject matter of determining whether a pointer field value of an existing data element is null, the Examiner refers to column 19, line 61 – column 20, line 30 of Garthwaite. Examiner's Answer, pages 12-13. It is noted that the claim language at issue here is "determining whether a pointer field value of the *existing data element*" is null. The Examiner ignores that the pointer field value must be for an existing data element. Instead, what the Examiner refers to in this portion of Garthwaite is simply how to obtain a table location using a field in a car's metadata structure. Nonetheless, this has nothing to do with determining whether a pointer field value of *an existing data element* is null. This is certainly so, as the recited determining of claim 13 is in the context of determining whether an existing data element is not a member of a data structure. As discussed above, Garthwaite also fails to disclose this subject matter. For all these reasons, the rejection of claim 13 should be reversed.

As to independent claim 21, the Examiner appears to contend that Garthwaite meets the recited subject matter of causing a branch to be taken in response to determining that a pointer value for an existing data element is not equal to an initialization value, where the branch is around one or more atomic operations to add the existing data element to a data structure. Examiner's Answer, page 14. Again, the Examiner relies on columns 19 and 20 of Garthwaite. Here, Garthwaite fails to teach or suggest anything with regard to an initialization value of a pointer value for an existing data element. Here, the Examiner contends that a null value is "similar to an initialization value". Examiner's Answer, page 14. Applicants disagree that a null value as disclosed in Garthwaite is similar to the recited initialization value. Instead, Garthwaite teaches that its null value "is the value that the old-table field had before replacement of the car's remembered-set table began." Garthwaite, column 20, lines 4-6. However, this has nothing to do with an initialization value of an existing data element. Instead, Garthwaite's null value simply identifies an old table field value before a replacement operation begins. Furthermore, this old table field is not a pointer value for an existing data element. Instead, Garthwaite simply teaches that its old table field is a pointer to a remembered set that is being replaced. Garthwaite, column 19, lines 60-64.

In addition, claim 21 recites that its branch taken in response to this determination is a branch around one or more atomic operations to add an existing data element to a data structure. Garthwaite fails in this regard as well. Here, the Examiner refers to blocks 232 and 233 of Garthwaite, which describe that an old table field's contents are taken as a table location if not null, and otherwise a linked list is followed. Garthwaite, column 20, lines 7-10. However, in no event does either of these operations cause a branch around one or more atomic operations to add an existing data element to a data structure. For all these reasons the rejection of claim 21 should be reversed.

Finally, it is unclear to Appellants what the Examiner is referring to on page 11 of the Answer with regard to a §101 rejection. As presently understood, there is no pending §101 rejection, as evidenced by the Examiner's agreement that the Grounds of Rejection to Be Reviewed on Appeal is correct. Examiner's Answer, page 7.

II. CONCLUSION

For the reasons set forth herein, as well as set forth in the Appeal Brief, Appellants respectfully request that each of the final rejections be reversed and that the claims subject to this Appeal be allowed to issue.

Respectfully submitted,

Date: May 21, 2010

/Mark J. Rozman/
Mark J. Rozman
Registration No. 42,117
TROP, PRUNER & HU, P.C.
1616 S. Voss Road, Suite 750
Houston, Texas 77057-2631
(512) 418-9944 [Phone]
(713) 468-8883 [Fax]
Customer No.: 21906